

REMARKS

Claims 1, 3-6, 8-11, 13-16, and 18-26 are pending in the present application. Claims 23-26 are added. Claims 1, 3, 4, 6, 11, 13, 14, 16, 21, and 22 are amended. Claims 1, 11, and 21 are amended to provide proper antecedent basis and to recite "creating, at a server, a component control file which defines enablement components needed to run applications from the server assigned to a user on a client and actions to be performed to install the enablement components, wherein the enablement components comprise required changes to an operating system of the client; creating, at the server, an installation control file which lists the applications that have been assigned to the user and the enablement components from the component control file which must be installed in order to execute the applications that have been assigned to the user on the client; determining, at the client, if the enablement components for applications listed in the installation control file are currently installed on the client; and if the enablement components for the applications listed in the installation control file are not currently installed on the client, installing, at the client, the enablement components for the applications listed in the installation control file on the client." These features are supported at least on page 11, lines 29-31 and page 20, lines 5-10 of the current specification.

Claims 3 and 13 are amended to incorporate the features of original claims 4 and 14. Claims 4 and 14 are amended to recite "if the enablement components for the applications listed in the installation control file are currently installed on the client, determining if versions of the enablement components for the applications listed in the installation control file are later than versions of the enablement components currently installed on the client; and if versions of the enablement components for the applications listed in the installation control file are later than versions of the enablement components currently installed on the client, installing the enablement components for the applications listed in the installation control file on the client." These features are supported at least on page 21, lines 21-30 of the current specification.

Claims 6, 16, and 22 are amended to recite "responsive to a request for updating a client, reading, at a client, a user's installation control file from a server which contains a

list of enablement components needed to run a set of applications from the server that have been assigned to the user of a client; determining, at the client, if each of the list of enablement components is installed on the client; and changing, at the client, an operating system of the client to install each of the list of enablement components that is not installed on the client." These features are supported at least on page 11, lines 29-31 and page 20, lines 5-10 and in Figure 5 of the current specification.

Claims 23 and 25 are added to recite "wherein the required changes to an operating system of the client include an application name assigned to the user, information files required to be installed on the client, registry files required to be installed on the client, directories required to be copied to the client, files required to be copied to the client, and commands required to be run on the client." These features are supported at least on page 13, line 8 to page 17, line 2 of the current specification.

Claims 24 and 26 are added to recite "wherein the step of changing an operating system of the client comprises: installing required information files on the client; installing required information files on the client; installing required registry files on the client; copying required directories to the client; copying required files to the client; and running required comments on the client." These features are supported at least on page 13, line 8 to page 17, line 2 of the current specification.

No new matter is added as a result of the above amendments. Reconsideration of the claims is respectfully requested.

I. Telephone Interview Summary

A telephone interview was conducted on August 15, 2005 with Examiner Barot with regard to features of independent claims 1 and 6. The Examiner indicated that based on the proposed amendments to the claims, rejection of claims 1, 6, 11, 16, 21, and 22 under 35 U.S.C. § 112, second paragraph should be overcome. In addition, Applicants' representative submitted that Wilde fails to teach the determining and installing steps in a manner as recited in claims 1 and 6. The Examiner indicated that the proposed amendments made to independent claims 1 and 6 potentially overcome the Wilde reference, but further search and consideration is required before an agreement can be reached with the Applicants' representative.

II. 35 U.S.C. § 112, Second Paragraph

The Office Action rejects claims 1, 3-6, 8-11, 13-16, and 18-22 under 35 U.S.C. § 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter, which Applicants regard as the invention. This rejection is respectfully traversed.

The Office Action states that there is insufficient antecedent basis for the limitations "the components" and "the applications" as recited in claims 1, 11, and 21. Specifically, "the applications" is unclear about the user specific application or run applications. Additionally, the locations of the both creating steps are unknown.

By this Response, claims 1, 11, and 21 are amended to recite "the enablement components" to provide proper antecedent basis for "the components". In addition, claims 1, 11, and 21 are amended to recite "the applications that have been assigned to the user on the client" and "applications listed in the installation control file" to provide the proper antecedent basis for "the applications." Furthermore, claims 1, 11, and 21 are amended to recite "creating, at a server, a component control file" and "creating, at the server, an installation control file" to specify the locations of both creating steps.

Furthermore, the Office Action states that it is unclear where the reading step as recited in claims 6, 16, and 22 is reading from and in response to. It is also unclear about the operating system of who/what. Additionally, the locations of both reading and changing steps are unknown. By this Response, claims 6, 16, and 22 are amended to recite "responsive to a request for updating a client, reading, at the client, a user's installation control file from a server" to clarify where the reading step is reading from and in response to and the location of the reading step. In addition, claims 6, 16, and 22 are amended to recite "changing, at the client, an operating system of the client" to clarify the operating system and the location of the changing step.

Therefore the rejection of claims 1, 6, 11, 16, 21, and 22 under 35 U.S.C. § 112, second paragraph has been overcome.

III. 35 U.S.C. § 102(e), Alleged Anticipation, Claims 1, 3-6, 8-11, 13-16, and 18-22

The Office Action rejects claims 1, 3-6, 8-11, 13-16, and 18-22 under 35 U.S.C. § 102(e) as being allegedly anticipated by Wilde et al (U.S. Patent No. 6,446,260). This rejection is respectfully traversed.

As to claim 1 the Office Action states:

As to claim 1, Wilde et al teach a method for updating client computers with user specific application enablement (see abstract; figures 1-2 and 4; and column 6 line 28 to column 7 line 36), comprising; creating a component control file which defines enablement components needed to run applications from a file server assigned a user on a client and actions to be performed to install the components, the components comprises required changes to an operating system of the client (figures 1-2; and column 7 line 37 to column 9 line 31); and creating an installation control file which lists the applications that have been assigned to the user and the components from the component control file which must be installed in order to execute the applications (figures 1-2 and 4; column 9 line 32 to column 10 line 62; and column 12 lines 15-48).

Office Action dated June 30, 2005, pages 3-4

A prior art reference anticipates the claimed invention under 35 U.S.C. § 102 only if every element of a claimed invention is identically shown in that single reference, arranged as they are in the claims. *In re bond*, 910 F.2d 831, 832, 15 U.S.P.Q.2d 1566, 1567 (Fed Cir. 1990). All limitations of the claimed invention must be considered when determining patentability. *In re Lowry*, 32 F.3d 1579, 1582, 21 U.S.P.Q.2d 1031, 1034 (Fed Cir. 1994). Anticipation focuses on whether a claim reads on the product or process a prior art reference discloses, not on what the reference broadly teaches. *Kalman v. Kimberly-Clark Corp.*, 713 F.2d 760, 218 U.S.P.Q. 781 (Fed. Cir. 1983). Applicants respectfully submit that Wilde does not teach each and every element of the claimed invention arranged as they are in claims 1, 11 and 21.

Amended independent claim 1, which is representative of claims 11 and 21 with regard to similarly recited subject matter, reads as follows:

1. A method for updating client computers with user specific application enablement, comprising:
creating, at a server, a component control file which defines enablement components needed to run applications from the server assigned to a user on a client and actions to be performed to install the enablement components, wherein the enablement components comprise required changes to an operating system of the client;

creating, at the server, an installation control file which lists the applications that have been assigned to the user and the enablement components from the component control file which must be installed in order to execute the applications that have been assigned to the user on the client;

determining, at the client, if the enablement components for applications listed in the installation control file are currently installed on the client; and

if the enablement components for the applications listed in the installation control file are not currently installed on the client, installing, at the client, the enablement components for the applications listed in the installation control file on the client. (Emphasis added).

Wilde does not teach the features emphasized above. As discussed in the Abstract, Wilde teaches a method and apparatus for providing personalization parameters to allow an operating system to install itself on a computer system with the provided personalization parameters. A first operating system personalization file is used by the operating system to configure itself. A personalization parameters file is provided and read by an operating system installation process which is initiated in the computer system. The operating system installation process then displays a graphical user interface from which a user is to select personalization parameters with which the operating system is to configure itself. An editing module is executed to edit the operating system configuration file to include at least a portion of the selected personalization parameters so that the operating system is configured with the selected personalization parameters when it installs itself.

However, Wilde does not teach determining, at the client, if the enablement components for applications listed in the installation control file are currently installed on the client. At column 12, line 49 to column 14, line 58, Wilde discloses:

Before deploying an OS in a workstation, the CM-user must log-on to the workstation. As shown in steps 114 and 116 of FIG. 5, a CM-user performs a service boot via workstation I/O 42. A "service boot" interrupts the normal workstation computer boot process to connect workstation 36 to the configuration manager either in the console 12 or, preferably, the server 14. As detailed in the Intel LANDesk® Configuration Manager Administrator's Guide, a service boot is performed in LCM by logging onto the client workstation 36, interrupting the normal boot process, and entering the CM-user's logon name and password. As noted in the Background section, this procedure preferably

accesses LANDesk® Service Agent ("LSA") 80 firmware located in an ethernet card installed on workstation 36 and provides for a connection between workstation 36 and server 14. Preferably, once connected to server 14, configuration manager modules are downloaded to workstation 36. Preferably, the configuration manager software downloaded by the server 14 into the workstation memory includes at least a logon module 51, a menu presentation module 52, and an OS installation module 54. Preferably, as detailed below, an OS setup program 68, and a post-OS installation module 90 are downloaded at a later time. Each of these modules is discussed below.

After service boot 116 has been performed, the CM-user must select an operating system to be deployed to workstation 36. To facilitate this, the logon module 51 reads the user name that was used to log on and reads users file 30 in server 14 to determine with which CM-user the user name is associated and which desktop profile file 34 is associated with the particular CM-user. Then, as shown in step 120 of FIG. 5, the logon module 51 sets a "comprofiles" environment variable 95 in workstation 36. The value of comprofiles environment variable 95 is the desktop profile file 34 assigned to the CM-user who has performed the service boot. As shown in step 122 of FIG. 6, which shows the steps implemented by the menu presentation module 52, the menu presentation module 52 reads the comprofiles environment variable and retrieves the services menu assigned to the particular CM-user from the services and menu files 32 in server 14. In step 124, the menu presentation module 52 displays the services menu 33 on monitor 49 as shown in FIG. 11. Display in menu format of information contained in a file is well known to those skilled in the art. The CM-user then selects from the service menu 33 the operating system to be deployed on workstation 36. At this point, the OS setup program 68 associated with the selected operating system is downloaded into workstation 36.

After selection of an operating system to be deployed, a GUI on the monitor 49 of workstation 36 is deployed to allow the CM-user to select a group of personalization parameters with which the selected operating system will be configured. The GUI advantageously allows a CM-user to quickly select a group of personalization parameters with which the selected operating system is to be configured without the need to manually edit an answer file or manually execute individual API's. To create and display the GUI, using the value from the comprofiles environment variable 95, the menu presentation module 52 reads the desktop profile 34 associated with the CM-user who is logged on and creates a menu item for each desktop profile included in the desktop profile 34. For example, if a desktop profile file 34 contained the section headings "[Computer 1]" and "[Computer 2]", then those two section headings would appear in the desktop profile menu. Then, as shown in step 128 and FIG. 12, the menu presentation module 52 displays the desktop profile menu 53. The desktop profile 34 from which desktop

profile menu 53 was generated would have the section heading "[Computer 1]", "[Computer 2]", and "[Computer 3]". As shown in step 130 the CM-user selects from the desktop profile menu 53 which desktop profile is to be used. The CM-user can do this using keyboard 47 or any other input device such as a mouse, trackball, or other pointing device. As shown in step 132, the menu presentation module 52 then stores the selection heading of the selected desktop profile in an "selectedcomp" environment variable 97 located in workstation 36.

After display of the GUI and selection of a group of personalization parameters with which the deployed operating system is to be configured, an editing module of the present invention places the personalization parameters from the selected group in generic answer file 64 to create an updated answer file 66. FIGS. 7 and 8 illustrate the steps of this editing process. Initially, in step 134, an editing module, referred to as an "OS installation module" 54, reads the value of compfiles environment variable 95 and then, in step 136 reads the value of selectedcomp environment variable 97. In steps 138 and 140, OS installation module 54 opens the desktop profile 34 designed in compfiles and retrieves the attributes of the selected desktop profile 39 designed in selectioncomp. In step 142, OS installation module retrieves and updates generic answer file 64 with the parameters identified in the selected desktop profile.

As shown in FIG. 13, to update the generic answer file 64, in step 310, OS installation module 54 reads the first key under the section head of the selected desktop profile 39. The installation module 54 automatically edits the generic answer file 64 with a portion of the personalization parameters in desktop profile 39. This advantageously avoids the need to manually edit generic answer file 64. In step 312, OS installation module 54 compares the first key under the section head of the selected desktop profile 39 with a list of parameters which can be placed in an answer file to determine whether the answer file can be edited with the value of the key. The list of keys which can be placed in an answer file is preprogrammed in OS installation module 54. If the value of the key can be placed in an answer file, OS installation module 54 reads the value of the key and looks for the matching key in generic answer file 64. In step 314, OS installation module 54 then places the value of the key in the argument of the matching key in generic answer file 64. In steps 316 and 318, if the key in the desktop profile is not the last key, OS installation module 54 moves the next key. If the key in the desktop profile is the last key, OS installation module 54 exists. If the key in the desktop profile is not a parameter which can be placed in answer file, in step 322 and 324, OS installation module 54 determines if it is the last key in the desktop profile and if it is not, module 54 moves on to the next key. If the desktop profile key is the last key, the OS installation module exists.

For example, if the first three lines of a selected desktop profile 39 appear as follows;

[Computer 1]

IP Address=123.123.123. 78

InitialPassword=jsmith

then OS installation module 54 will read the key "IP Address"; check to see if this key is one with which an answer file can be configured; locates the "IP Address" key in generic answer file 64; and place the value 123.123.123.78 in the argument of the "IP Address" key in generic answer file 64. OS installation module will then move on to the "InitialPassword" key. It will discover that this key cannot be configured in an answer file, thus, it will move on to the next key or exit if "InitialPassword" is the last key in desktop profile 39.

After moving through all the keys in desktop profile 39, OS installation module exists to step 144. In step 144, the OS installation module 54 stores an update answer file 66 in workstation 36. In step 146, the OS installation module 54 places the values of comprofiles environment variable 95 (the selected desktop profile file) and selectedcomp environment variable 97 (the selected desktop profile index) in an initialization file "continue.ini" 72 to be used for post-OS install personalization as detailed below.

Column 12, line 49 to column 14, line 58.

As can be seen, Wilde teaches that before deploying an operating system in a workstation, the user first log on to the workstation, performs a service boot to connect the workstation to a configuration manager, which includes an OS installation module and an OS setup program. After the service boot, the user selects an operating system to be deployed, and a group of personalization parameters with which the operating system will be configured. An editing module then places the personalization parameters in a generic answer file to form an updated answer file. At column 14, line 59 to column 15, line 10, Wilde teaches:

At this stage, as show in FIGS. 4 and 8, the installation of the OS into the workstation CPU 38 is initiated by an OS setup program 68 located in workstation 36. In step 150 of FIG. 8, which shows the steps completed by OS setup program 68, OS setup program 68 reads the compressed OS files 70 from server 14 associated with the service selected by the CM-user, and in step 152, decompresses the OS files 70. As shown in step 154, the OS being installed then reads the updated answer 66 file from the workstation 36 which was stored there by the OS installation module 62 and, in step 156 installs using the parameters from the updated answer file 66. As noted above, because the OS being installed is programmatically designed to use an answer file during

installation, steps 154 and 156 are built into the OS installation program 68. Also as noted above, details of the use and operation of setup programs for Windows®NT, Windows®95 and Windows®98 is provided in, respectively, "Microsoft® Windows®NT Workstation Operating System Deployment Guide," "Microsoft® Windows®95 Resource Kit," and "Microsoft® Windows®98 Resource Kit," each of which has been incorporated by reference.

Column 14, line 59 to column 15, line 10.

In the above section, Wilde teaches that after the answer file is updated, the OS setup program initiates the installation of the operating system by reading compressed OS files from a server, decompresses the OS files. The OS being installed then reads the updated answer file and installs the OS using parameters from the updated answer file. Thus, instead of first determining if enablement components for applications in the installation control file are currently installed on the client, Wilde reads the updated answer file with the personalization parameters and automatically installs the operating system using the parameters. Nowhere in the reference does Wilde teach determining, at the client, if the operating system or any other component in the updated answer file is currently installed on the workstation before installing the operating system. Therefore, Wilde does not teach determining if the enablement components for applications listed in the installation control file are currently installed on the client, as recited in claims 1, 11, and 21 of the present invention.

Since Wilde does not teach determining, at the client, if the enablement components for applications listed in the installation control file are currently installed on the client, Wilde does not and would not teach if the enablement components for the applications listed in the installation control file are not currently installed on the client, installing, at the client, the enablement components for the applications listed in the installation control file on the client. Wilde is only concerned with simplifying OS deployment by allowing a user, such as an administrator, to include personalization parameters in an updated answer file, such that the operating system may be deployed to a plurality of workstations without having to manually edit the answer file for each deployment. Wilde is not concerned with determining, at the client, if enablement components for applications that are assigned to the user are currently installed on the

client, such that even if the enablement components are not currently installed, the applications may still be executed on the client by installing the enablement components that are needed to run applications. Since Wilde neither concerns with enablement components that are needed to run applications that are assigned to the user, nor with installing the enablement components if they are not currently installed on the client in order to run the applications, Wilde does not and would not teach each and every feature of claims 1, 11, and 21 of the present invention.

With regard to amended independent claim 6, which is representative of claims 16 and 22 of the present invention, Wilde also does not teach determining, at the client, if each of the list of enablement components is currently installed on the client. The Examiner alleges in the Office Action that Wilde teaches these features in Figures 5-9. However, as discussed above in the arguments presented for claims 1, 11, and 21, Wilde teaches reading an updated answer file which includes personalization parameters selected by the user and installing operating system using those parameters. Wilde neither teaches nor suggests that a determination is made on the client as to whether each of the list of enablement components for the operating system is installed on the client. To the contrary, Wilde teaches a system that deploys operating systems to a plurality of workstations without regard to what is currently installed on the workstations. Therefore, Wilde does not teach determining, at the client, if each of the list of enablement components is installed on the client, as recited in claims 6, 16, and 22 of the present invention.

In addition, Wilde does not teach changing, at the client, an operating system of the client to install each of the list of enablement components that is not installed on the client. Since Wilde does not teach determining, at the client, if each of the list of enablement components is installed on the client, Wilde would not teach changing an operating system of the client to install the non-installed component. Wilde changes the operating system of the client in order to install the operating system using personalization parameters selected by the user. Wilde does not change the operating system of the client in order to install component that is not currently installed on the client. To the contrary, Wilde installs the operating system on the client regardless of whether the component of the operating system is currently installed. Therefore, Wilde

does not teach changing, at the client, an operating system of the client to install each of the list of enablement components that is not installed on the client, as recited in claims 6, 16, and 22 of the present invention.

In view of the above, Applicants respectfully submit that Wilde does not teach the features of claims 1, 6, 11, 16, 21, and 22. At least by virtue of their dependency on claims 1, 6, 11, and 16 respectively, Wilde does not teach or suggest the features of dependent claims 3-5, 8-10, 13-15, 18-20, and 23-26.

In addition, Wilde does not teach the specific features of claims 3-5, 8-10, 13-15, 18-20, and 23-26 of the present invention. For example, with regard to dependent claim 4, which is representative of dependent claim 14 with regard to similarly recited subject matter, Wilde does not teach if the enablement components for the applications listed in the installation control file are currently installed on the client, determining if versions of the enablement components for the applications listed in the installation control file are later than versions of the enablement components currently installed on the client; and if versions of the enablement components for the applications listed in the installation control file are later than versions of the enablement components currently installed on the client, installing the enablement components for the applications listed in the installation control file on the client.

As discussed above in arguments presented for claims 1, 6, 11, 16, 21, and 22, Wilde does not teach determining if enablement components are currently installed on the client. Therefore, Wilde would not teach if the enablement components for the applications listed in the installation control file are currently installed on the client, determining if versions of the enablement components for the applications listed in the installation control file are later than versions of the enablement components currently installed on the client. Wilde does not even mention anything about versions of the operating system, let alone enablement components of applications that are assigned to the user. At column 9, lines 55-67, Wilde teaches generic parameters that the user may personalized, which include target path to write OS files, keyboard layout, display type, and drivers, pointing type and drivers. In addition, the user may personalize username, domain name, and IP address in the updated answer file. However, Wilde does not teach comparing the enablement components versions for the applications listed in the

installation control file with the enablement components versions that are currently installed in order to determine which versions are later. Therefore, Wilde does not teach the features of claims 4 and 14 of the present invention.

In addition, since Wilde does not teach comparing the versions between the enablement components for applications listed in the installation control file with the enablement components currently installed on the client, Wilde would not teach if versions of the enablement components for the applications listed in the installation control file are later than versions of the enablement components currently installed on the client, installing the enablement components for the applications listed in the installation control file on the client. To the contrary, Wilde does not teach making any determination between versions of enablement components before installing the operating system. Rather, Wilde installs the operating system as soon as the updated answer file is read. Therefore, Wilde does not teach the features of claims 4 and 14 of the present invention.

Therefore, the rejection of claims 3-5, 8-10, 13-15, 18-20, and 23-26 under 35 U.S.C. § 102(e) has been overcome.

IV. Newly Added Claims, Claims 23- 26

Newly added claims 23 - 26 have been added to the application. These claims are patentable over the cited references for the same reasons as independent claims from which they depend.

Further with respect to claims 23 and 25, Wilde does not teach that the required changes to the operating system include an application name assigned to the user, information files required to be installed on the client, registry files required to be installed on the client, directories required to be copied to the client, files required to be copied to the client, and commands required to be run on the client. In Figure 4, Wilde teaches files such as user files, which includes user information (column 9, lines 31-32); services and menu files, which includes a plurality of service menus that the user uses when performing a service boot (column 9, lines 10-20); desktop profile files, which include individual user profile containing end-user personalization parameters such as username, login name, computer name, domain name, and IP addresses, etc. (column 10,

lines 51-55); generic answer files, which includes generic parameters, such as target path to write OS files, keyboard layout, display type and drivers, pointing type and drivers ; and compressed OS files, which are compressed files. However, nowhere in the reference does Wilde mentions required changes that includes required information files to be installed on the client, directories to be copied to the clients, or commands to be run on the client. Therefore, Wilde does not teach the features of newly added claims 23 and 25 of the present invention.

With regard to claims 24 and 26, Wilde does not teach installing required information files on the client; installing required registry files on the client; copying required directories to the client; copying required files to the client; and running required comments on the client. As discussed above in the arguments presented for claims 23 and 25, Wilde fails to teach required changes that include required information files, directories to be copied to the clients, or commands to be run on the client. Therefore, Wilde would not teach installing these required changes in the manner that is stated in claims 24 and 26.

V. Conclusion

It is respectfully urged that the subject application is patentable over Wilde and is now in condition for allowance.

The Examiner is invited to call the undersigned at the below-listed telephone number if in the opinion of the Examiner such a telephone conference would expedite or aid the prosecution and examination of this application.

DATE: 8/31/05

Respectfully submitted,



Wing Yan Mok
Reg. No. 56,237
Yee & Associates, P.C.
P.O. Box 802333
Dallas, TX 75380
(972) 385-8777
Agent for Applicants